ADLxMLDS-hw4

孫奧 R05922147

# Model description (2%)

## model structure

總的架構是這樣的：有一個Generator，和一個Discriminator。Generator的輸入是noise的condition拼接在一起的vector，輸出的是一張(64,64,3)的矩陣(我們將這個矩陣理解成圖片)。而Discriminator的輸入是(64,64,3)的矩陣，輸出是一個實數值(如果不在0~1會讓他通過sigmoid)，將其解釋成是否屬於真實分佈的幾率的判斷。

## objective function

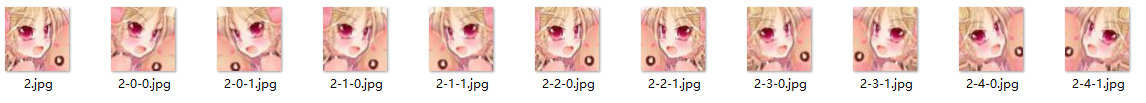
對於G而言很簡單，他的objective function是想要讓D(G)更接近與1，這邊用cross entropy來完成。

而對於D而言，在這個作業中來自3個部分：Discriminator對於真實資料的判斷、對於G生成資料的判斷、對於錯誤的label(condition)的判斷。

具體實作的是將各個值與相對應的0/1做cross entropy。

# How to improve your performance (2%)

其一就是增加資料的方法。將原本的一張照片做5種角度的旋轉，以及把所有的照片做鏡像翻轉。



如上圖所示，一張照片變成了11張，這樣資料集的大小也變大了11倍。通過增加資料的數量得到的結果

其二在於sample的部分。由於這次作業需要餵給錯誤的label以及label對應錯誤的圖片(這邊想說一下：原始的論文中並不需要錯誤的圖片，而在討論和網上找到的code當中都新增了這一個(正確label,錯誤圖片)的error)，我使用的是每過10個EPOCH重新sample這些pairs（一開始嘗試每次都重新sample訓練的會很滿甚至很難訓練起來）

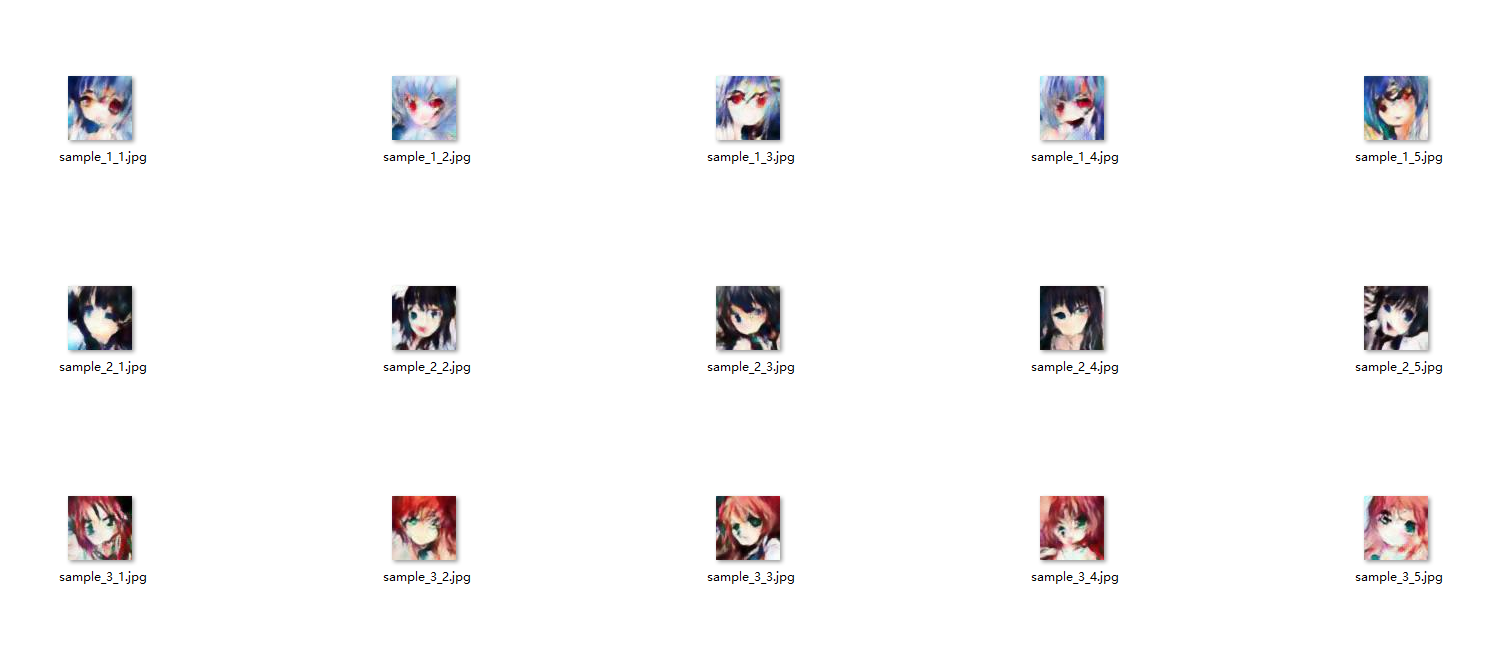
# Experimental settings and observation (2%)

## Experimental settings

hp = {  
 'Z\_shape': [100],  
 'R\_shape': [64, 64, 3],  
 'C\_shape': [23],  
 'beta1': 0.5,  
 'beta2': 0.9,  
 'lr': 0.00001,  
 'batch\_size': 128,  
 'g\_output\_activation': 'sigmoid',  
 'batch\_norm': *True*}

其中Z\_shape代表noise的維度，C\_shape代表one-hot的condition的維度(這邊一共有23維：12種頭髮concat上11種的眼鏡),beta1和beta2是AdamOprimizer的參數。

## Observation



這是挑選一下seed所產生的結果圖。個人覺得如果說這是人臉我還是相信的。並且顏色之類的也能對上，效果還是挺不錯的。